ESPECIALIZACIÓN EN CIENCIA DE DATOS

TRABAJO FINAL INTEGRADOR

DETECCIÓN DE ACTIVIDAD PESQUERA UTILIZANDO DATOS AIS CON LIGHTGBM

Alumno: Nicolás Agustín Marcovecchio

Título de grado: Ingeniero en electrónica.

Profesora:
Gambini María Juliana

Ciudad de Buenos Aires, 22 de julio de 2022.

Dedicatoria.

A Dámaris, y nuestra hija por nacer Isabella.

Tabla de contenidos.

1. Introducción.

El sistema AIS (*Automatic Identification System*) sirve para evitar colisiones a partir de que un buque transmite su posición a los demás. En la actualidad todavía hay buques pesqueros que no apagan sus sistemas al hacer pesca ilegal, y al ser un equipo el cual se configura manualmente muchos buques no se identifican como pesqueros o clonan la identificación de otro buque (Global Fishing Watch, Spoofing: One Identity Shared by Multiple Vessels, 2016).

La actividad pesquera puede proporcionar a las autoridades, los investigadores y a los políticos información para tener una imagen más completa de la pesca y a la sostenibilidad de los recursos marinos.

Lo que se busca en este trabajo es mejorar el estado actual para detectar esta actividad.

Se utilizara el novedoso conjunto de datos, ya que muchos trabajos anteriores se los ve limitado en este aspecto, publicado por Global Fishing Watch (GFW) en el 2020.
Este trabajo incluye el tratamiento para remover los datos faltantes y atípicos, resolver la granularidad temporal (los datos AIS en la realidad no se captan a intervalos constantes), la generación de un vector de características y el entrenamiento de un modelo óptimo de clasificación utilizando el método *lightgbm* (Microsoft Corporation, 2022) para comparar con lo alcanzado actualmente por GFW en (Kroodsma et al., 2018).

2. Estado de la cuestión.

Más de 400.000 barcos transmiten sus ubicaciones cada año a través del AIS (Global Fishing Watch, What is AIS?, 2020).

Desde el año 2012 a la actualidad hubo un incremento exponencial de satélites que captan tráfico AIS por la reducción de costos de poner un satélite en el espacio. No solo son capaces agencias espaciales con millones de dólares, sino que cualquier compañía con miles de dólares ahora puede poner en órbita un satélite. La cobertura mundial está cada vez más completa y se espera que el volumen y la importancia de estos datos se incrementen mucho más en los siguientes años, por lo cual es relevante desarrollar herramientas efectivas y eficientes de aprendizaje automático para trabajar este tipo de datos.

La IUU, siglas del inglés que significan la pesca ilegal, no reportada, y no regulada (Fisheries and Oceans Canada, 2019), es la amenaza más seria a la sustentabilidad de la pesca en el mundo, y se estima que esta equivale al 30% de la pesca mundial con un daño de 23 mil millones de dólares anuales.
Más de mil millones de personas dependen de la pesca como recurso principal de proteína, el 33% de las zonas de pescas mundiales están sobre saturadas, y uno de cada cinco peces son atrapados ilegalmente o de forma no regulada según la *United Nations Food & Agriculture Organization*.

El 90% de los grandes peces han desaparecido principalmente por la sobrepesca según un censo del 2010 de la vida marina.

GFW es una organización que intenta concientizar sobre el problema de pesca ilegal, y trabaja para hacerlo visible.

AIS (International Telecommunication Union, 2014), es un sistema que permite a los buques comunicar su posición, y otra información como su curso, identificación, tipo de buque, entre otros. Fue diseñado para evitar colisiones.
Utiliza dos frecuencias este sistema, los canales marinos 87B (161.975 MHz), y el 88B (162.025 MHz). Consiste en una modulación GMSK (*Gaussian minimum shifting key*), sobre canales de 25 o 12.5 kHz utilizando el protocolo *High-level Data Link Control* (HDLC).
S-AIS se refiere a las transmisiones capturadas vía satélites. Actualmente hay una constelación de satélites que se dedican a capturar este tráfico, el cual está en constante crecimiento teniendo cada vez más datos y con menor latencia.

El estándar AIS es obligatorio para los buques sometidos al Convenio *SOLAS* (*Safety of Life at Sea*), con las siguientes características:

- Buques con arqueo bruto superior a 500 GT.
- Buques en viaje internacional con arqueo bruto superior a 300 GT.
- Todos los buques de pasajeros, independientemente de su tamaño.

El convenio SOLAS es el más importante de todos los tratados internacionales sobre la seguridad de los buques.

Hay una directiva europea 2002/59/E1, que está en fase de aprobación, al cual hará obligatorio el uso de AIS para los buques de pesca con el siguiente calendario de aplicación:

- Pesqueros entre 24 y 45 metros de eslora: no más tarde que 3 años desde la entrada en vigor.
- Pesqueros entre 18 y 24 metros de eslora: no más tarde que 4 años desde la entrada en vigor.
- Pesqueros entre 15 y 18 metros de eslora: no más tarde que 5 años desde la entrada en vigor.
- Pesqueros de nueva construcción de más de 15 metros de eslora: no más tarde que 18 meses desde la entrada en vigor.

El problema de clasificación de buques pesqueros tiene al menos tres perspectivas que pueden ser utilizadas (Hu et al., 2016)

- Predicción estructurada:
  La detección de pesca puede ser vista como un problema de etiquetar secuencias en la cual la entrada es una serie de trayectorias del buque, y la salida es una secuencia de etiquetas (está pescando, o no).
- Clasificación en serie de tiempos:
  Desde esta perspectiva el problema se transforma en un mapeo a series de tiempo, en el cual la similitud de dos secuencias es medido.
- Reconocimiento de imágenes:
  Desde este enfoque es visto como un problema de segmentación de imagen en el cual la trayectoria es segmentada en segmentos de "pescando" y "no pescando"

GFW clasifica los buques de pesca en 40 clases, los cuales 16 son buques de pesca según se observa en la **Figura 1**.

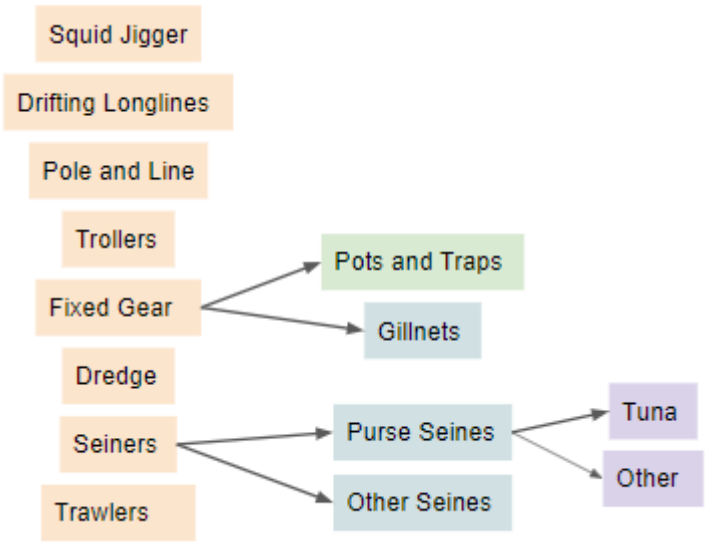

Figura 1. Clasificación de buques de pesca. Fuente GFW.

La principal investigación de la cual se derivan las demás es la presentada por (De Souza et al., 2016), en la cual utilizaron datos AIS para detectar buques pesqueros. Parte de los autores trabaja para GFW.
Este trabajo también expone el patrón en el cual un buque pesquero, según su tipo, puede ser identificado mientras pesca.
Este trabajo describe los patrones que se observaron en los datos para los siguientes tipos de buque:

- *Trawler*: Involucra el arrastre de una o más redes detrás del buque ya sea al nivel del suelo marino, o en la columna de agua. Mientras pescan generalmente el buque desacelera y trata de mantener una velocidad constante para mantener la tensión en la red lo más tensa posible. Esto puede durar unos minutos o unas horas según la densidad de la presa. Lo típico es que dure entre 3 y 5 horas. La velocidad en el cual realizan esta maniobra varía entre los 2.5 y 5.5 nudos. La Figura 2 muestra el método de pesca utilizado por los Trawlers.

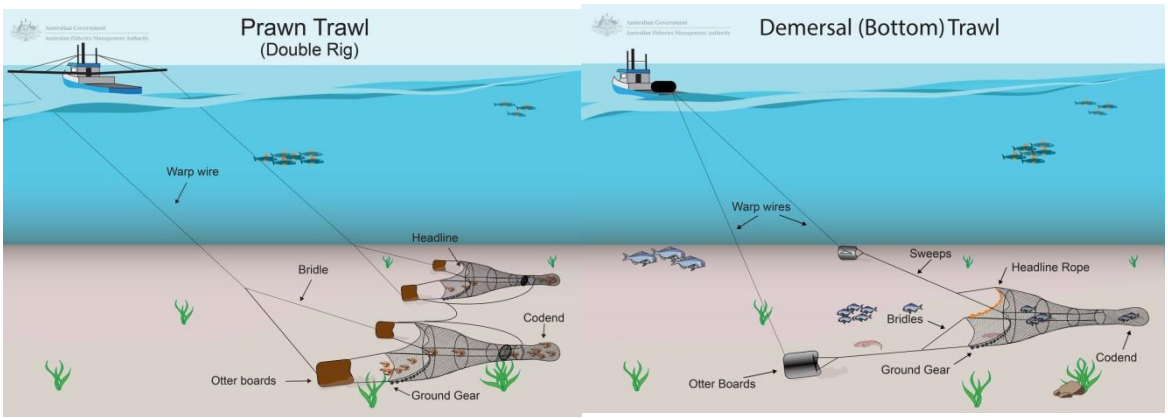

Figura 2. Método de pesca de Trawlers. Fuente Afma

- *Longliner*: Involucra poner líneas de pesca (de hasta varios kilómetros de longitud) equipadas con cientos de anzuelos. Las líneas están dispuestas a varias profundidades con el uso de flotadores. Para tirar la línea el buque se desplaza a una velocidad ligeramente menor a su velocidad habitual de desplazamiento. Luego de que el último anzuelo es tirado en el agua la línea es dejada unas horas. Durante este tiempo el buque aprovecha para tirar otras líneas o simplemente continúa su trayecto a baja velocidad con la línea enganchada. Por ultimo regresa por el mismo recorrido levantando la línea pudiendo conllevar todo este trabajo hasta un día. Generalmente la velocidad es constante, pero puede ir variando según la pesca y el número de personal trabajando. La media encontrada en el estudio de 16 buques fue de 6.5 horas. La Figura 3 muestra el método de pesca utilizado por los Longliner.

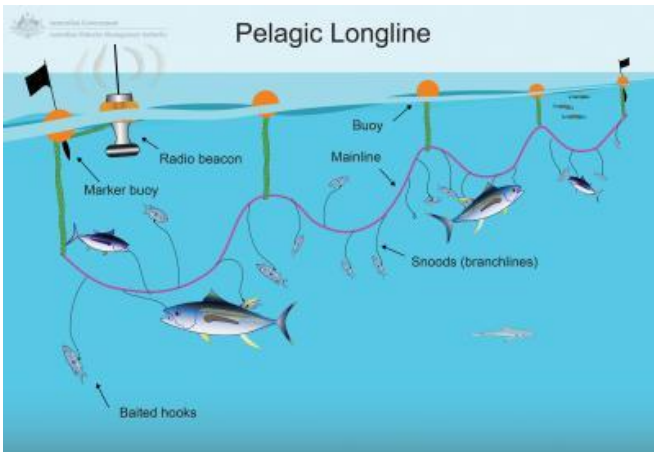

Figura 3. Método de pesca de Longliner. Fuente Afma

- *Purse Seiner*: Involucra largas redes las cuales van colgadas verticalmente agarradas de un flotador alrededor de cardúmenes de peces. Para evitar que los

peces escapen la red debe ser puesta rápidamente y a grandes velocidades (alrededor de 10 nudos). Una vez que la red encierra en un círculo al cardumen, el fondo de la red se cierra y la red se iza. La duración de este proceso puede variar de una a varias horas. La Figura 4 muestra el método de pesca utilizado por los Purse Seine.

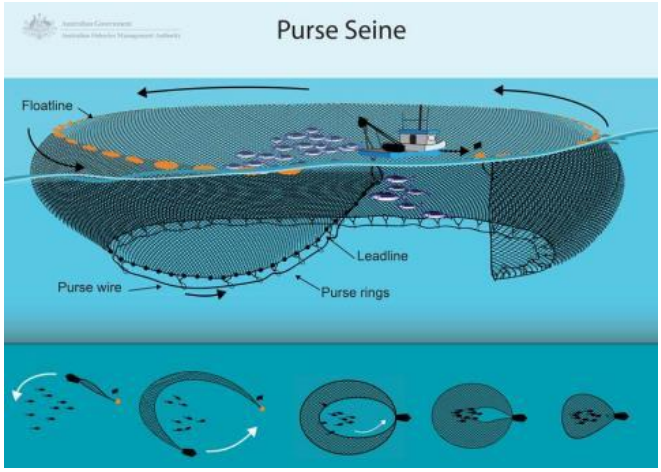

Figura 4. Método de pesca de Purse Seine. Fuente Afma

Adicionalmente a los descriptos en este trabajo se encuentran:

- Fixed Gear: Colocan trampas estacionarias o redes, las cuales generalmente las dejan 24 horas, y luego las pasan a recoger. La **Figura 5** muestra el método de pesca utilizado por los Fixed Gear.

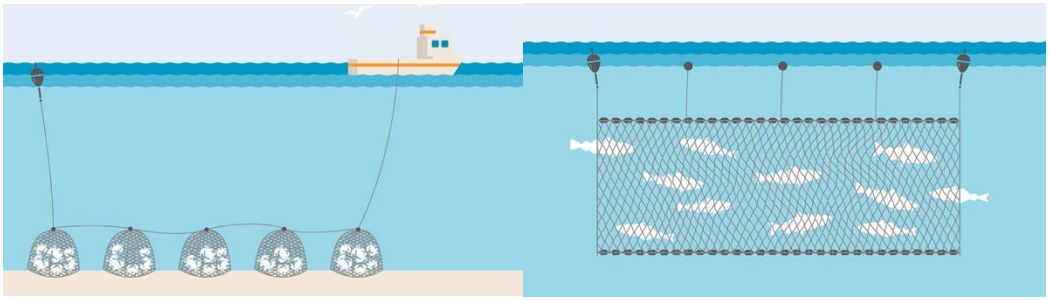

Figura 5. Método de pesca de Fixed Gear. Fuente Marine Stewardship council.

- Trollers: Son buques que pescan utilizando anzuelos. La **Figura 6** muestra el método de pesca utilizado por los Trollers.

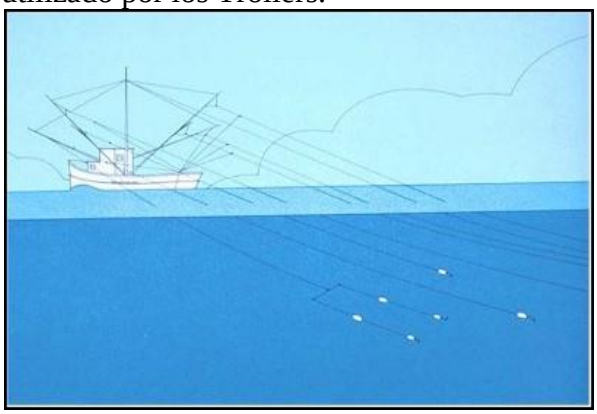

Figura 6. Método de pesca de Trollers. Fuente QCS.

El trabajo (De Souza et al., 2016) utilizaron HMM (*Hidden Markov Model*) para resolver la clasificación. Algo a destacar es que para el L*ongliner* no alcanzo simplemente con la velocidad, utilizó HMM pero previamente el algoritmo de segmentación de Lavielle´s, y también un segundo algoritmo para detectar si el segmento está compuesto de curvas. Para el *Purse Seiner* se tuvo en cuenta que la mayoría no pesca de noche, y que su patrón consiste en dos actividades principales. Primero poner la red para luego moverse a grandes velocidades, lo cual es una actividad corta y puede que no esté representada en los datos por la falta de granularidad (insuficiencia de cobertura satelital), luego sigue el acarreo y achique que esto puede llevar hasta varias horas, y esto es lo que se utiliza para detectar la actividad. Este trabajo expone gráficos de probabilidad en función de la velocidad para estos 3 casos teniendo en cuenta lo mencionado anteriormente, y claramente se ven estos patrones. Para resolver la problemática se usaron datos AIS obtenidos desde enero del 2011 a octubre de 2015 de 83 T*rawlers*, 16 *Longliners*, 7 *Purse Seiners*, siendo estos etiquetados previamente por un experto.

Cronológicamente a continuación (Jiang et al., 2016) usaron *Restricted Boltzmann Machines* para desarrollar *autoencoders* con *backpropagation* y un ventaneo. Para tener en cuenta que los intervalos son irregulares hicieron un submuestreo para reducir las variaciones y el ruido entre distintos puntos. Por ejemplo, si una serie de puntos están dentro de un intervalo de 100 segundos solo una muestra es seleccionada. Las longitudes y latitudes se utilizaron relativas, y generaron una matriz lo cual es una imagen de la trayectoria interpolada. Luego los resultados los compararon con SVM (*Summary Vector Machines*), y RF (*Random Forests*). Como futuro trabajo propusieron usar RNN (*Recurrent Neural Networks*) para tomar ventaja de la información temporal. Como datos utilizaron solo de *Longliners* los cuales no se ven disponibles.

Las RNN fueron puestas a prueba en (Shen et al., 2020), aprovechando que los datos son secuenciales. Este algoritmo tiene lazos de realimentación el cual es una memoria a corto plazo, esto significa que cada entrada no tiene solo el resultado de la capa oculta previa, sino el valor predicho con anterioridad. Cómo vector de características no utilizaron los valores absolutos de latitud y longitud, sino que usaron las posiciones relativas entre puntos consecutivos, sino llevaba el modelo a hacer *overfitting*. Como la actividad pesquera está altamente correlacionada con la velocidad del buque y el cambio en su curso utilizaron el SOG (*Speed over ground*), y el COG (*Course over ground*). Como el SOG es un valor instantáneo el cual puede causar errores cuando es usado para juzgar comportamientos a largo plazo se consideró también la diferencia de tiempo entre muestras, distancia, y velocidad promedio entre puntos.
Como datos utilizaron AIS de la costa de Taiwán de T*rawlers*, *Trolling*, y *Longliner*.

Un enfoque con CRF (*Conditional Random Fields*) fue investigado en (Hu et al., 2016), el cual es un algoritmo popular para resolver problemas de predicción estructurados, como categorizar secuencias en procesamiento de lenguaje natural. Para la discretización trabajaron en bins, para el vector de características utilizaron la longitud y latitud como diferencial (con el punto anterior), el COG, el SOG, y el estado previo. Como trabajo a futuro se proponen a investigar mejores maneras de desarrollar características adicionales como densidad y ángulo.
Como datos utilizaron información recogida de 14 L*ongliners* desde el 1 de junio del 2012 al 31 de diciembre del 2013.

Como último trabajo observado hasta el momento es el prestantado por (Arasteh et al., 2020) el cual realizaron un modelo utilizando CNN (*convolutional neural network*), en el cual reconstruyen la trayectoria para identificar el recorrido del buque, generando un vector de características basado en este movimiento el cual es invariante a la localización y el tiempo. Generaron un ventaneo de a segmentos, y finalmente el estado del buque en cada punto temporal se determinó en función de las etiquetas de la mayoría en el segmento. Este algoritmo luego fue comparado con un perceptrón multicapa, RF, y XGBoost.
Utilizaron datos provistos por GFW de *Purse Seiner*, L*ongliners*, T*rawlers*, y *Fixed Gear*.

Actualmente GFW utiliza CNN (Kroodsma et al., 2018). Ellos previamente filtran los datos removiendo puntos físicamente imposibles (velocidad no realística entre puntos), los segmentos menores a 5 posiciones, y si hay gaps mayores a 24 horas se crea un punto intermedio artificial. Se modificó el conjunto de datos para que haya al menos un punto cada 5 minutos, y se generaron 12 vectores de características por punto.
Utilizaron trayectos de 146 *Drifting Longlines*, 5 P*ole and Line*, 36 P*urse Seines*, 9 S*et Gillnets*,4 S*et Longlines*, 37 T*rawlers*, y 3 T*rollers* para entrenar. Fueron casi 174.000 horas (503 MMSI) utilizadas para el entrenamiento y el resto para el testeo.
En el entrenamiento utilizaron TensorFlow sobre Google's Cloud ML con 5 instancias de GPUs en paralelo.
La efectividad alcanzada fue de:

Clase de Buque	Precision	Recall	Accuracy	F1
Longlines	0.92	0.94	0.91	**0.93**
Purse Seines	0.78	0.81	0.95	**0.79**
Fixed Gear	0.95	0.88	0.97	**0.9**
Trawlers	0.98	0.94	0.96	**0.96**

Tabla 1. M*odelo actual empleado por GFW. Fuente* (Kroodsma et al., 2018)

Todos los trabajos presentados con anterioridad presentan un enfoque de deep learning e incluso algunos solo entrenan para un solo tipo de pesquero.

3. Definición del problema.

En el mundo muchos datos AIS son transmitidos incompletos, por lo tanto, muchos buques pesqueros no se identifican como tales, lo que implica una dificultad para las autoridades locales de identificar la pesca ilegal para determinar el nivel de pesca en la zona. Esto ya sea porque están realizando pesca ilegal o la carga del sistema no es realizada de manera correcta.

Poder identificar la actividad pesquera automáticamente a partir de datos AIS nos brinda una ayuda más para tener un panorama más completo de la pesca a nivel global para la sustentabilidad de este recurso.

4. Justificación del trabajo.

Desde el 24/03/2020 está disponible públicamente en GFW la descarga de los conjunto de datos para resolver este tipo de problemas para los siguientes pesqueros y etiquetada por expertos (*Anonymized AIS Training Data*, 2020):

- unknown.csv (801.05 MB)
- trollers.csv (18.04 MB)
- trawlers.csv (496.56 MB)
- purse_seines.csv (185.12 MB)
- pole_and_line.csv (17.33 MB)
- fixed_gear.csv (171.79 MB)
- drifting_longlines.csv (1.65 GB)

Los trabajos realizados con anterioridad, no resuelven o entrenan para todos los casos expuestos en el conjunto de datos de más arriba, en algunos casos resolviendo solo para un tipo de pesquero. En (Arasteh et al., 2020) utilizaron algoritmos de minería de datos para comparar con la CNN generada, pero no se los ve optimizados.

Se buscara mejorar los resultados alcanzado actualmente por GFW (Kroodsma et al., 2018).

5. Alcances del trabajo y limitaciones.

Los resultados obtenidos servirán para las organizaciones que estudian y vigilan el IUU, como una posible forma de mejorar sus algoritmos de identificación de pesca. Estaremos limitados por el Hardware disponible (CPU Intel Core i9-10900F; RAM: 32gb).

A menos que se indique lo contrario, los datos de GFW tienen una licencia de Creative Commons Attribution-ShareAlike 4.0 International y un código bajo una licencia de Apache 2.0.

6. Hipótesis.

Es posible predecir si un buque está pescando o no a partir de datos AIS.

Contamos con las siguientes variables:

- Posición:
  - Tipo: Variable independiente, cuantitativa.
  - Definición nominal: La posición es un sitio que ocupa un cuerpo, susceptible de determinarse por coordenadas espaciales.
- Velocidad:
  - Tipo: Velocidad: Variable dependiente, cuantitativa.
  - Definición nominal: La velocidad es la magnitud física de carácter vectorial que relaciona el cambio de posición con el tiempo.
- Rumbo:
  - Tipo: Variable independiente, cuantitativa.
  - Definición nominal: El rumbo es la dirección que sigue o ha de seguir una embarcación.
- Tiempo:
  - Variable independiente, cuantitativa.
  - Definición nominal: El tiempo es una magnitud física con la que se mide la duración o separación de acontecimientos.
- Estado:
  - Variable dependiente, cualitativa clasificatoria
  - Definición nominal: Etiqueta del conjunto de datos que me indica si está pescando en ese punto, o no.

Relación: Hipótesis de multivariables.

Relación planteada: Causa efecto.

Definición de las variables operacionales:
Velocidad, Curso, Tiempo: La velocidad, curso, y tiempo de un buque puede ser adquirida a partir del sistema AIS (Sistema de identificación automática) para buques que cuentan con este dispositivo.

7. Objetivos.

7.1. Objetivo general.

Desarrollar un modelo de clasificación que para un trayecto determinado de un buque (puntos AIS) pueda predecir si es un buque se encuentra pescando o no en un punto dado, intentando mejorar lo alcanzado actualmente por GFW (Kroodsma et al., 2018).

7.2. Objetivo específico.

- Estudiar en profundidad el conjunto de datos, y hacer un análisis del mismo.
- Sacar datos incompletos y atípicos.
- Hacer un remuestreo del conjunto de datos para tener un punto cada cierto intervalo, y normalizar las variables.
- Generar un vector de características.
- Generar un modelo utilizando lightgbm, y optimizar los hiperpárametros utilizando una optimización bayesiana.
- Comparar con los resultados obtenidos por GFW (Kroodsma et al., 2018).

8. Metodología.

8.1 Técnicas.

Para el análisis exploratorio de los datos se utilizará pandas para encontrar aquellos datos atípicos, repetidos, o faltantes. Haremos gráficos para corroborar las relaciones entre variables con la variable a predecir.
Se realizara un programa para remover datos atípicos, normalizar las variables, resolver la granularidad temporal (remuestreo), y generar un vector de características en el conjunto de datos.
Se aplicara un modelo de ligthgbm con una optimización bayesiana de hiperpárametros, y con el resultado se armara una comparación con lo alcanzado actualmente por GFW para alcanzar los objetivos planteados.
Todos estos programas se encuentran disponibles en el siguiente enlace:
https://github.com/nmarcovecchio/tfi_itba_gfw

8.2 Herramientas.

Como herramientas se usaran aquellas de código abierto.
Para el análisis de datos se utilizara pandas, y matplotlib.
Como base de datos seleccionaremos Postgres para generar una base de datos georreferenciada, y como visor de puntos AIS se usara QGIS.
Para realizar los modelos se optara por utilizar scikit-learn.

8.3 Análisis de los datos.

Para facilitar nuestro trabajo se ha utilizado la librería papermill (nteract team, 2018) la cual es una herramienta para parametrizar y ejecutar múltiples Jupyter Notebook en paralelo para los diferentes conjunto de datos.

En la **Tabla 2** se observa que nos encontramos frente a un caso de datos desbalanceados. La clase minoritaria es justamente lo que nos interesa predecir, cuando el buque está pescando. Como ejemplo, para el caso de los Longlines aproximadamente el 1% de los puntos corresponde a un buque pescando.

Clase de Buque	Cantidad de buques	Cantidad de puntos	Puntos no pescando	Puntos pescando
Longlines	110	13.968.727	13.748.986	138.163
Purse Seines	28	1.545.323	1.522.474	2.740
Fixed Gear	35	1.559.137	1.517.279	10.665
Trawlers	49	4369101	4.191.707	61.930
Trollers	5	166.243	158.398	2.966

Tabla 2. Cantidad de buques, y puntos.

En la **Tabla 3** se observa que tenemos datos repetidos, y faltantes. Para el caso de los Longlines, la cantidad de datos repetidos es importante y debemos removerlos ya que no nos brinda información a nuestro modelo.

Clase de Buque	Repetidos	NA
Longlines	2.796.639	98
Purse Seines	30.126	7
Fixed Gear	48.320	0
Trawlers	180.497	78
Trollers	577	0

Tabla 3. Cantidad de datos repetidos y faltantes.

En la **Tabla 4** se observa la velocidad promedio de cada clase de buque cuando se encuentran pescando:

Clase de Buque	Velocidad promedio pescando [mph]
Longlines	5,31
Purse Seines	4,04
Fixed Gear	3,51
Trawlers	4,32
Trollers	4,18

Tabla 4. Velocidad promedio de cada clase de buque pescando.

En la **Figura 7** se observa que tan espaciados se encuentran los datos AIS temporalmente. Para todos los tipos de buque, excepto los Longlines, se ve como el promedio de transmisión ocurre cada 12, 15, o 18 minutos. En el caso de los Longlines hay un gran pico cerca del cero al tener muchos datos repetidos.

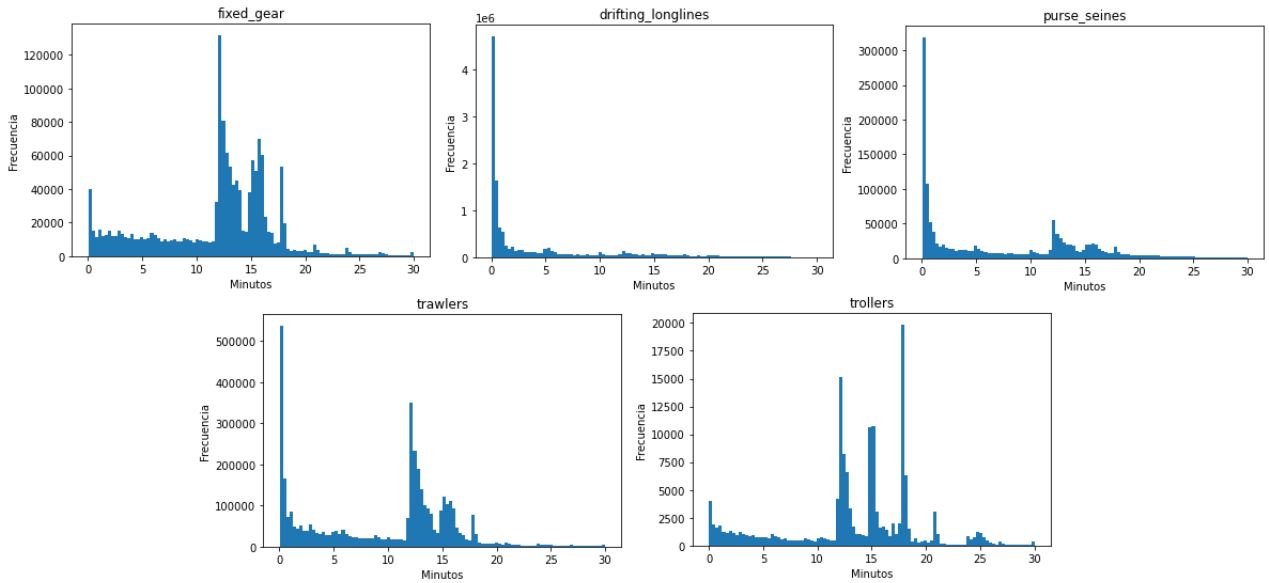

Figura 7. Diferencia de tiempo en minutos entre transmisiones consecutivas.

En la **Figura 8** se observa como los Longlines vuelven sobre el camino recorrido, y los Purse Seines tienen patrones circulares.
Para la construcción de estas imágenes se ha utilizado el software QGIS sobre el conjunto de datos en crudo. Se generó un vector de líneas entre puntos agrupados por MMSI (cada buque tiene su recorrido independiente), y ordenados con el timestamp para poder graficar el recorrido.

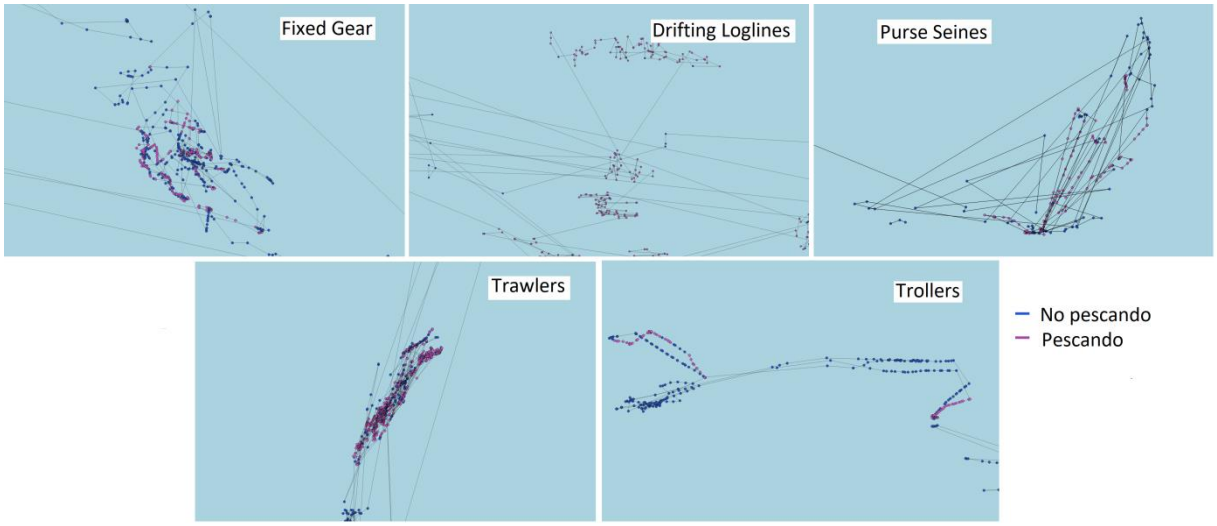

Figura 8. Recorridos de buques pescando.

Figura 9. Matriz de correlacion Fixed Gear.

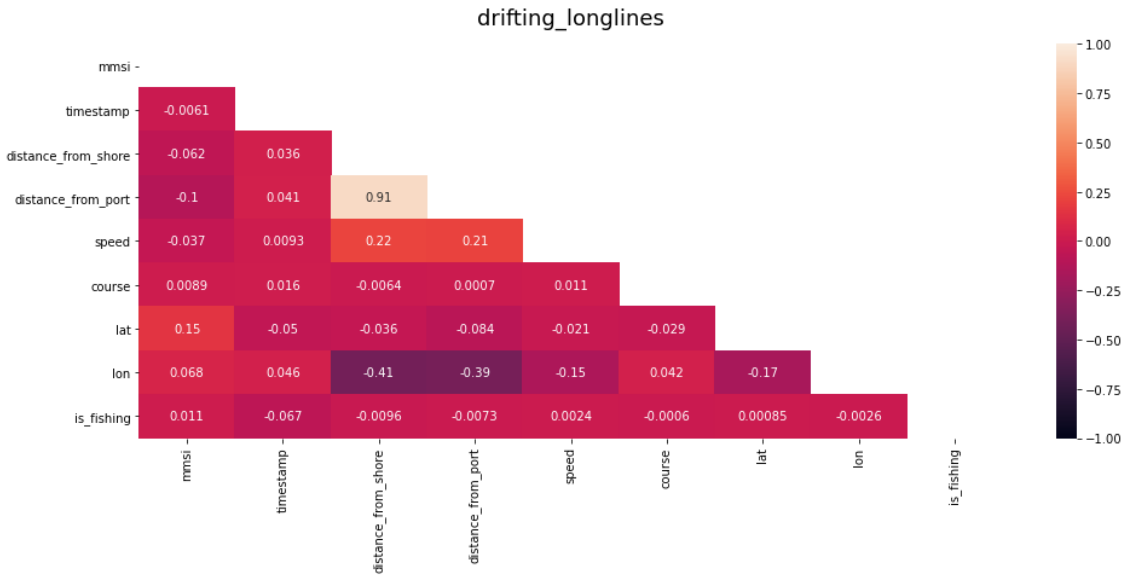

Figura 10. Matriz de correlacion longlines.

Figura 11. Matriz de correlacion Purse Seines.

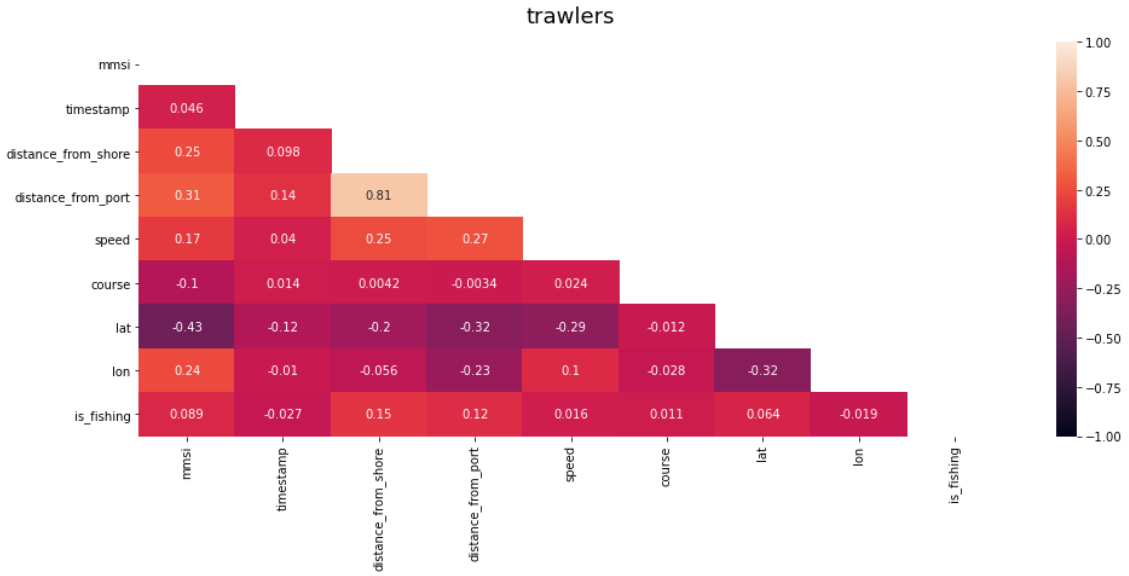

Figura 12. Matriz de correlacion Trawlers.

Figura 13. Matriz de correlacion Trollers.

En la **Figura 9, 10, 11, 12, y 13** se observa las matrices de correlación para los diferentes tipos de buques.
Hay una gran correlación positiva entre la distancia a la costa, y la distancia al puerto, esto se debe a que son directamente proporcionales.
También hay una correlación positiva entre la variable a predecir con la distancia al puerto, y la distancia a la costa para los Trawlers, y los Trollers.

8.4 Vector de características.

Hacemos un remuestreo para quedarnos con un solo punto cada 15 minutos, ya que el promedio de transmisiones consecutivas sucede entre los 12 y 18 minutos.
Tomamos como valor de ventaneo 7 muestras en el pasado para que cada punto contenga la información de las últimas 2 horas, el cual es un valor acorde con (Arasteh et al., 2020) "2 horas es lo suficientemente informativo y representativo del comportamiento del movimiento de un buque" (p.5).
Los puntos a menos de 3 millas náuticas de la costa son descartados.
Nos quedamos con la mitad de puntos pescando, y la otra mitad no pescando haciendo un problema balanceado.
Los puntos en el cual el buque no se encuentra pescando serán elegidos aleatoriamente.

La variable **latitud** y **longitud** son convertidas a radianes.

$$Latitud = \frac{Latitud * \pi}{180.0} \qquad Longitud = \frac{Longitud * \pi}{180.0}$$

La variable **curso** es normalizada.

$$Course = \frac{Course}{360.0}$$

Timediff contiene la diferencia de tiempo en segundos entre puntos consecutivos.

$$Timediff = t2 - t1$$

Teniendo en cuenta que para uso náutico se considera que es de noche cuando el sol se encuentra por debajo de los -12° con respecto a la línea de horizonte, con la latitud, longitud, y timestamp se genera una variable llamada "**sun_state**" representando si es de día o de noche. En (De Souza et al., 2016) llegaron a la conclusión de que la mayoría de los purse seiners no pesca de noche. Para lograr esto se ha utilizado ephem (Downey, 1998), el cual es un paquete para realizar cálculos astronómicos de alta precisión.

La variable **Distancia** representa la diferencia en metros entre dos puntos consecutivos.

Calcularemos la velocidad rectilínea **S0**, la aceleración **A0**, la sobre aceleración **J0**, y la derivada del curso **C0**.

$$S0 = \frac{Distancia}{Timediff}; \quad A0 = \frac{S0}{Timediff}; \quad J0 = \frac{A0}{Timediff}; \quad C0 = \frac{c2-c1}{Timediff}$$

Vavg y **Cavg** representan la velocidad promedio y el curso promedio entre puntos consecutivos.

$$Vavg(x2,x1) = \frac{Speed(x2) + Speed(x1)}{2}; Cavg(x2,x1) = \frac{Course(x2) + Course(x1)}{2}$$

Delta_S y **Delta_C** representan la variación de velocidad y curso entre puntos consecutivos.

$$Delta\_S(x2,x1) = \frac{Speed(x2) - Speed(x1)}{2}; Cavg(x2,x1) = \frac{Course(x2) - Course(x1)}{2}$$

Cada punto de entrenamiento contiene las siguientes columnas:

['speed', 'course', 'lat', 'lon', 'is_fishing', 'datetime','timediff', 'sun_state','distance', 'S0', 'A0', 'J0', 'C0', 'Vavg', 'Delta_S', 'Cavg', 'Delta_C', 'speed_lag_1', 'speed_lag_2', 'speed_lag_3', 'speed_lag_4', 'speed_lag_5', 'speed_lag_6', 'speed_lag_7', 'course_lag_1', 'course_lag_2', 'course_lag_3','course_lag_4', 'course_lag_5', 'course_lag_6', 'course_lag_7','S0_lag_1', 'S0_lag_2', 'S0_lag_3', 'S0_lag_4', 'S0_lag_5', 'S0_lag_6', 'S0_lag_7', 'A0_lag_1', 'A0_lag_2', 'A0_lag_3', 'A0_lag_4', 'A0_lag_5', 'A0_lag_6', 'A0_lag_7', 'J0_lag_1', 'J0_lag_2', 'J0_lag_3', 'J0_lag_4', 'J0_lag_5', 'J0_lag_6', 'J0_lag_7', 'C0_lag_1', 'C0_lag_2', 'C0_lag_3', 'C0_lag_4', 'C0_lag_5', 'C0_lag_6', 'C0_lag_7', 'Delta_C_lag_1', 'Delta_C_lag_2', 'Delta_C_lag_3', 'Delta_C_lag_4', 'Delta_C_lag_5', 'Delta_C_lag_6', 'Delta_C_lag_7', 'Delta_S_lag_1', 'Delta_S_lag_2', 'Delta_S_lag_3', 'Delta_S_lag_4', 'Delta_S_lag_5', 'Delta_S_lag_6', 'Delta_S_lag_7', 'Vavg_lag_1', 'Vavg_lag_2', 'Vavg_lag_3', 'Vavg_lag_4', 'Vavg_lag_5', 'Vavg_lag_6', 'Vavg_lag_7']

Luego del remuestreo cada 15 minutos, en la **Figura 14** se observa como han quedado espaciados temporalmente los puntos AIS. Ahora el pico máximo se encuentran centrado en alrededor de los 15 minutos.

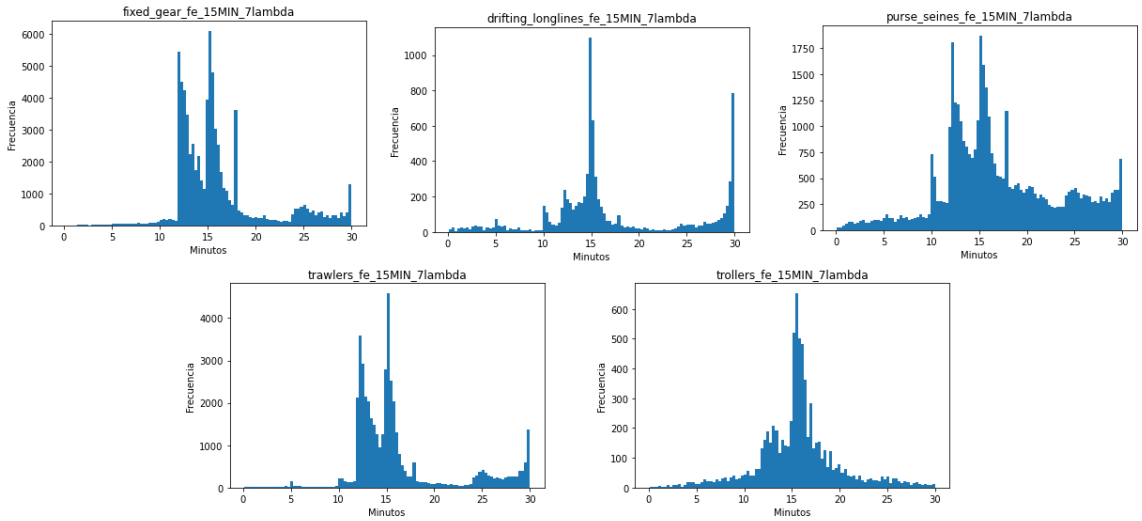

Figura 14. Diferencia de tiempo en minutos entre transmisiones consecutivas luego del remuestreo cada 15minutos.

Se observa en la **Tabla 5** la cantidad de puntos que nos hemos quedado por tipo de buque para generar nuestro modelo.
Esperamos tener más puntos, ya que otros trabajos se han quedado con aproximadamente entre 10.000 y 20.000 puntos para generar los modelos.
Comparando con la **Tabla 2**, de 138.163 puntos pescando para los Longlines, nos hemos quedado con 2.313 puntos pescando. Esto se debe a que hay una gran cantidad de transmisiones poco espaciadas temporalmente.

Buque	Puntos no pescando	Puntos pescando
Longlines	2313	2313
Purse Seines	283	283
Fixed Gear	2414	2414
Trawlers	5357	5357
Trollers	722	722

Tabla 5. Cantidad de puntos luego del remuestro, balanceo, y generación de características.

8.5 Resultados.

Para las 5 clases de buque entrenadas se corrió los algoritmos clásicos de minería de datos con su búsqueda de hiperpárametros por cuadricula (Grid search), y luego se utilizó lightgbm con optimización bayesiana con 2000 iteraciones para encontrar el mejor modelo.
El 80% de los datos se utilizó para entrenamiento, y el 20% restante para testeo.
Lo marcado en amarillo en las tablas de comparación de modelos para el caso de búsqueda de hiperpárametros por cuadricula corresponden a los mejor parámetros encontrados.
La **Tabla 6, 7, 8, 9, y 10** muestran los resultados conseguidos para cada tipo de buque, y modelo. Comparando los F1 se observa como lightgbm sobrepasa a los modelos clásicos de minería de datos, y para el caso de los Purse Seines lo iguala. Algo a tener en cuenta que para el caso de los Purse Seines nos hemos quedado con 566 puntos, el valor más pequeño comparado con los demás casos.
En la **Figura 18, 20, 22, 24, y 26** se observa como la optimización bayesiana va encontrando el espacio óptimo de cada parámetro, y el RMSE tiende a disminuir a lo largo de las iteraciones.
Con un valor de RMSE entre 0.2 y 0.5 podemos decir que el modelo puede predecir los datos con relativa precisión.
En la **Tabla 6** podemos observar el RMSE obtenido por cada buque para el modelo lightgbm.
Para todos los modelos, SVM obtuvo los peores resultados.
Las variables más importantes que se observan en la **Figura 15, 17, 19, 21, y 23** son para todos los casos la latitud y longitud. Se puede decir que hay zonas donde es más probables encontrar un buque pescando.

Buque	RMSE
Longlines	0.55
Purse Seines	0.44
Fixed Gear	0.50
Trawlers	0.37
Trollers	0.25

Tabla 6. Comparación de RMSE, modelo lightgbm.

Longlines						
Modelo	Ajuste de Hiperparámetros	Hiperparámetros	Prec.	Rec.	Acc.	F1
Random Forest	GridSearch	• bootstrap: [True, False] • max_depth: [12,13,14,15,16,17,18,20,25] • max_features: ["auto", "sqrt", "log2"] • min_samples_leaf: [1,2,3,4,5,6,7,8] • min_samples_split: [2, 5 , 10, 20] • n_estimators: [100, 200,400,800,1000] • cv: 3	0.653	0.719	0.651	0.684
SVM sigmoid	GridSearch	• gamma: [1e-2,1e-3,1e-4,1e-5,1e-6] • C: [0.001,0.01,0.1,1,10,100] • coef0: [0.01,0.1,1,10]	0.593	0.583	0.570	0.588
SVM linear	GridSearch	• C: [0.0001,0.001,0.01,0.1,1,10,100,1000]	0.591	0.563	0.565	0.576
SVM rbf	GridSearch	• gamma:[1e-2,1e-3,1e-4,1e-5,1e-6] • C: [0.001,0.01,0.1,1,10,100]}	0.595	0.628	0.580	0.611
LGBM	Opt. Bayesiana	• Best value (rmse): 0.55104 • Best params: • n_estimators: 100 • learning_rate: 0.09250387921080122 • num_leaves: 158 • max_depth: 11 • min_data_in_leaf: 12 • lambda_l1: 0.008967982124002516 • lambda_l2: 0.07603455563636284 • min_gain_to_split: 2.4985547882562664 • feature_fraction: 0.9	0.72	0.74	0.71	0.73

Tabla 6. Comparación de modelos para Longlines.

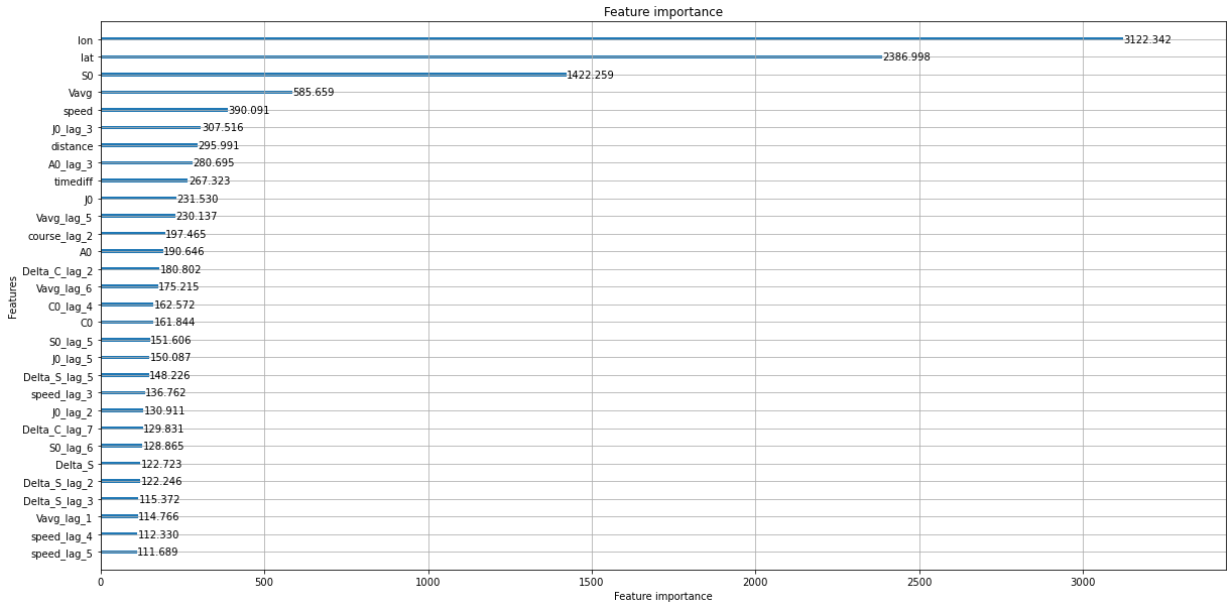

Figura 15. Importancia de las variables para Longlines.

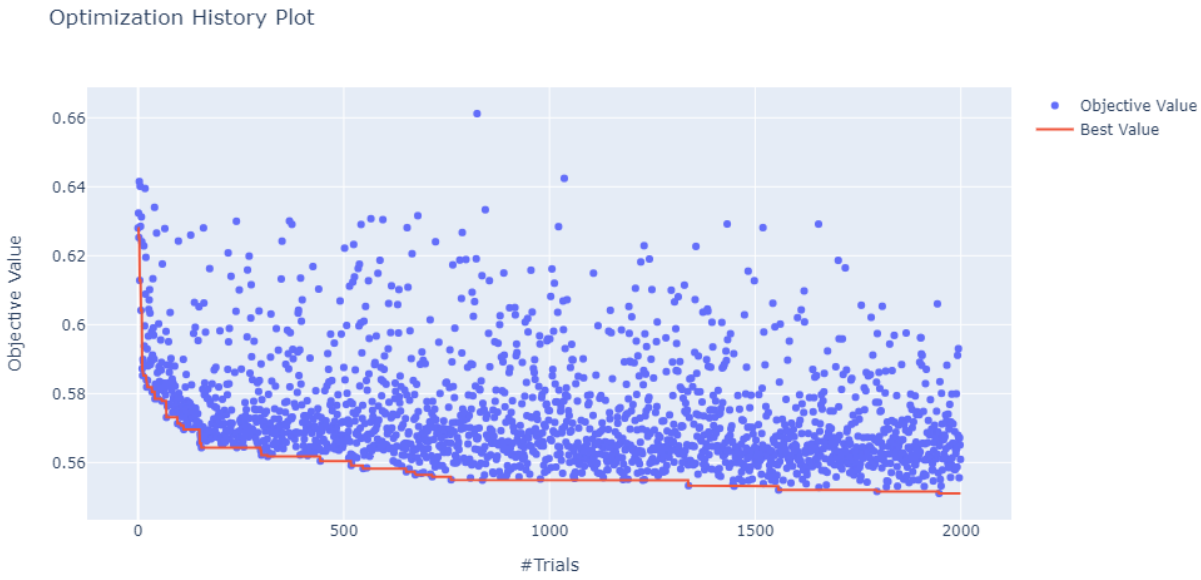

Figura 16. RMSE a lo largo del entrenamiento para Longlines.

Fixed Gear						
Modelo	Ajuste de Hiperparámetros	Hiperparámetros	Prec.	Rec.	Acc.	F1
Random Forest	GridSearch	• bootstrap: [True, False] • max_depth: [12,13,14,15,16,17,18,20,25] • max_features: ["auto", "sqrt", "log2"] • min_samples_leaf: [1,2,3,4,5,6,7,8] • min_samples_split: [2, 5 , 10, 20] • n_estimators: [100, 200,400,800,1000] • cv: 3	0.696	0.715	0.693	0.705
SVM sigmoid	GridSearch	• gamma: [1e-2,1e-3,1e-4,1e-5,1e-6] • C: [0.001,0.01,0.1,1,10,100] • coef0: [0.01,0.1,1,10]	0.561	0.543	0.546	0.551
SVM linear	GridSearch	• C: [0.0001,0.001,0.01,0.1,1,10,100,1000]	0.569	0.561	0.555	0.565
SVM rbf	GridSearch	• gamma:[1e-2,1e-3,1e-4,1e-5,1e-6] • C: [0.001,0.01,0.1,1,10,100]}	0.568	0.587	0.557	0.577
LGBM	Opt. Bayesiana	• Best value (rmse): 0.43380 • Best params: • n_estimators: 100 • learning_rate: 0.09042525565056131 • num_leaves: 118 • max_depth: 11 • min_data_in_leaf: 18 • lambda_l1: 0.1947738553741819 • lambda_l2: 0.042329227305151394 • min_gain_to_split: 0.0044752172288633485 • feature_fraction: 0.7	0.736	0.729	0.726	0.732

Tabla 7. Comparación de modelos para Fixed Gear.

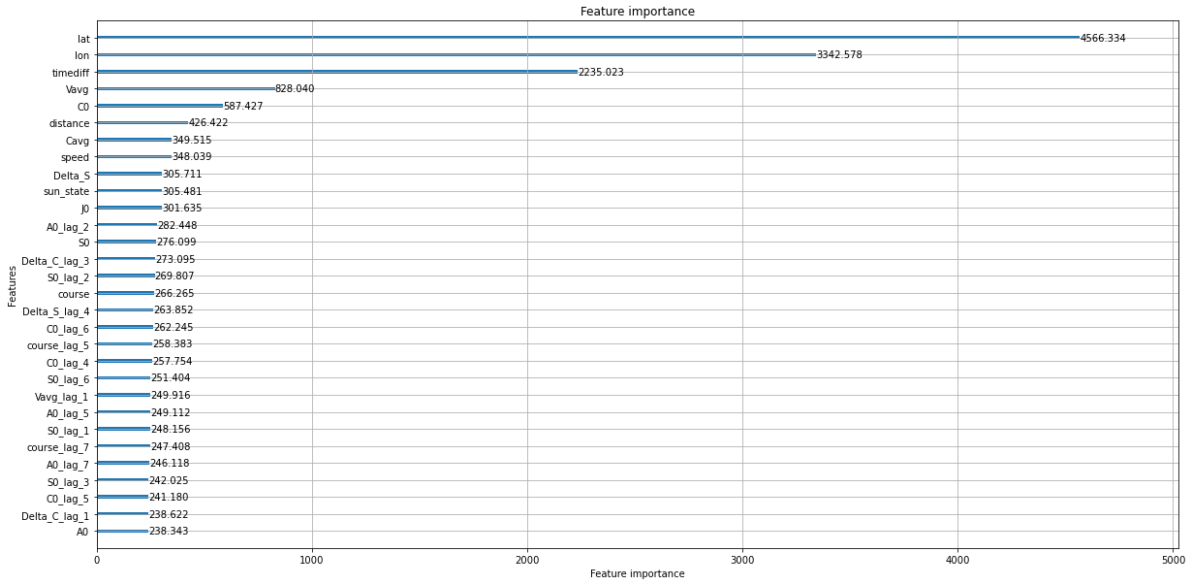

Figura 17. Importancia de las variables para Fixed Gear.

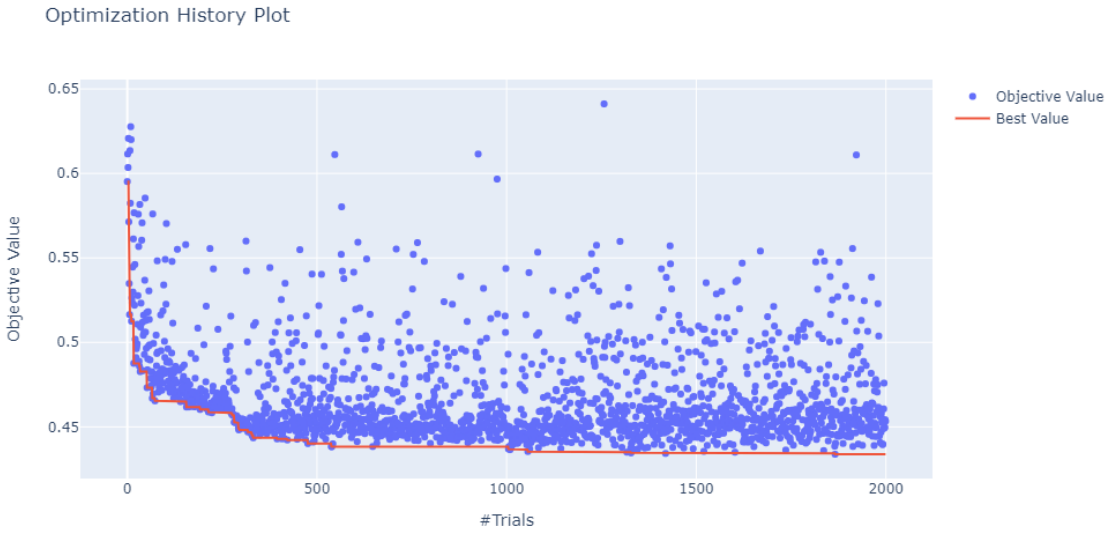

Figura 18. RMSE a lo largo del entrenamiento para Fixed Gear.

Purse Seiners						
Modelo	Ajuste de Hiperparámetros	Hiperparámetros	Prec.	Rec.	Acc.	F1
Random Forest	GridSearch	• bootstrap: [True, False] • max_depth: [12,13,14,15,16,17,18,20,25] • max_features: ["auto", "sqrt", "log2"] • min_samples_leaf: [1,2,3,4,5,6,7,8] • min_samples_split: [2, 5 , 10, 20] • n_estimators: [100, 200,400,800,1000] • cv: 3	0.75	0.69	0.74	0.72
SVM sigmoid	GridSearch	• gamma: [1e-2,1e-3,1e-4,1e-5,1e-6] • C: [0.001,0.01,0.1,1,10,100] • coef0: [0.01,0.1,1,10]	0.61	0.64	0.63	0.62
SVM linear	GridSearch	• C: [0.0001,0.001,0.01,0.1,1,10,100,1000]	0.66	0.67	0.68	0.67
SVM rbf	GridSearch	• gamma:[1e-2,1e-3,1e-4,1e-5,1e-6] • C: [0.001,0.01,0.1,1,10,100]}	0.65	0.67	0.67	0.66
LGBM	Opt. Bayesiana	• Best value (rmse): 0.49987 • Best params: • n_estimators: 100 • learning_rate: 0.20279307651408265 • num_leaves: 188 • max_depth: 12 • min_data_in_leaf: 22 • lambda_l1: 0.20698016744188447 • lambda_l2: 14.404319657780741 • min_gain_to_split: 0.0035439504792846026 • feature_fraction: 0.5	0.77	0.67	0.75	0.72

Tabla 8. Comparación de modelos para Purse Seiners.

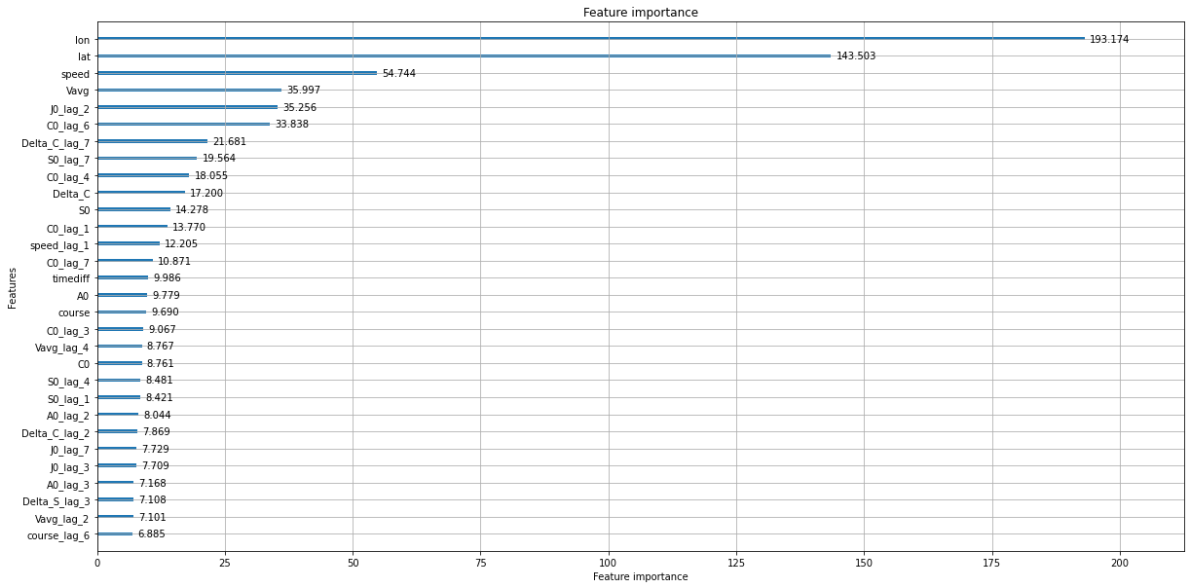

Figura 19. Importancia de las variables para Purse Seiners.

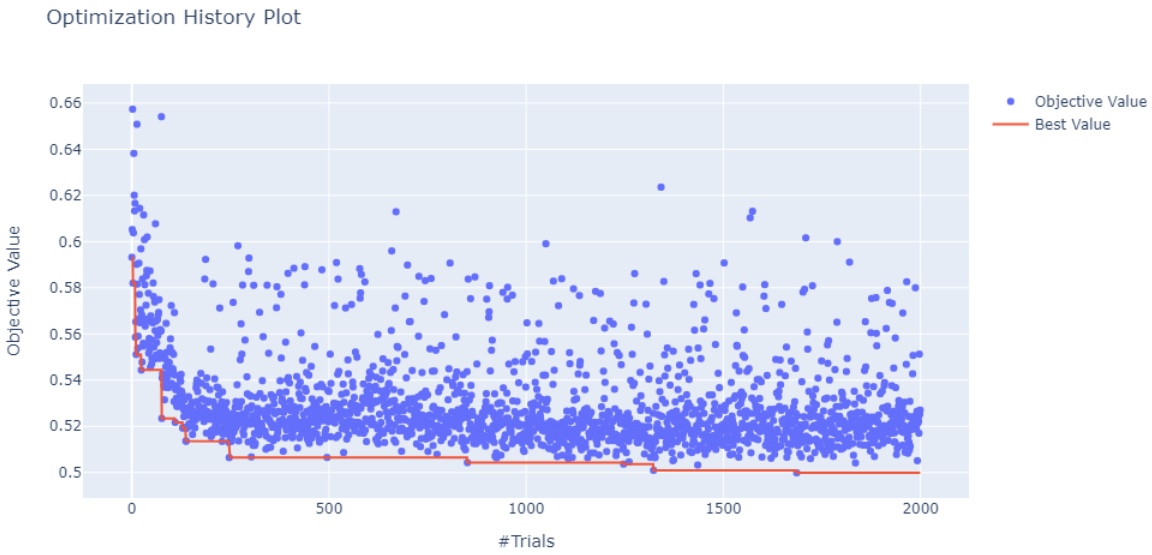

Figura 20. RMSE a lo largo del entrenamiento para Purse Seiners.

Trawlers						
Modelo	Ajuste de Hiperparámetros	Hiperparámetros	Prec.	Rec.	Acc.	F1
Random Forest	GridSearch	• bootstrap: [True, False] • max_depth: [12,13,14,15,16,17,18,20,25] • max_features: ["auto", "sqrt", "log2"] • min_samples_leaf: [1,2,3,4,5,6,7,8] • min_samples_split: [2, 5 , 10, 20] • n_estimators: [100, 200,400,800,1000] • cv: 3	0.85	0.69	0.78	0.76
SVM sigmoid	GridSearch	• gamma: [1e-2,1e-3,1e-4,1e-5,1e-6] • C: [0.001,0.01,0.1,1,10,100] • coef0: [0.01,0.1,1,10]	0.58	0.73	0.60	0.64
SVM linear	GridSearch	• C: [0.0001,0.001,0.01,0.1,1,10,100,1000]	0.60	0.74	0.62	0.66
SVM rbf	GridSearch	• gamma:[1e-2,1e-3,1e-4,1e-5,1e-6] • C: [0.001,0.01,0.1,1,10,100]}	0.65	0.63	0.64	0.64
LGBM	Opt. Bayesiana	• Best value (rmse): 0.37185 • Best params: • n_estimators: 100 • learning_rate: 0.04311202304096241 • num_leaves: 648 • max_depth: 12 • min_data_in_leaf: 10 • lambda_l1: 0.012333564903524673 • lambda_l2: 0.047466085872233964 • min_gain_to_split: 1.8052058684195835 • feature_fraction: 0.9	0.82	0.82	0.82	0.82

Tabla 9. Comparación de modelos para Trawlers.

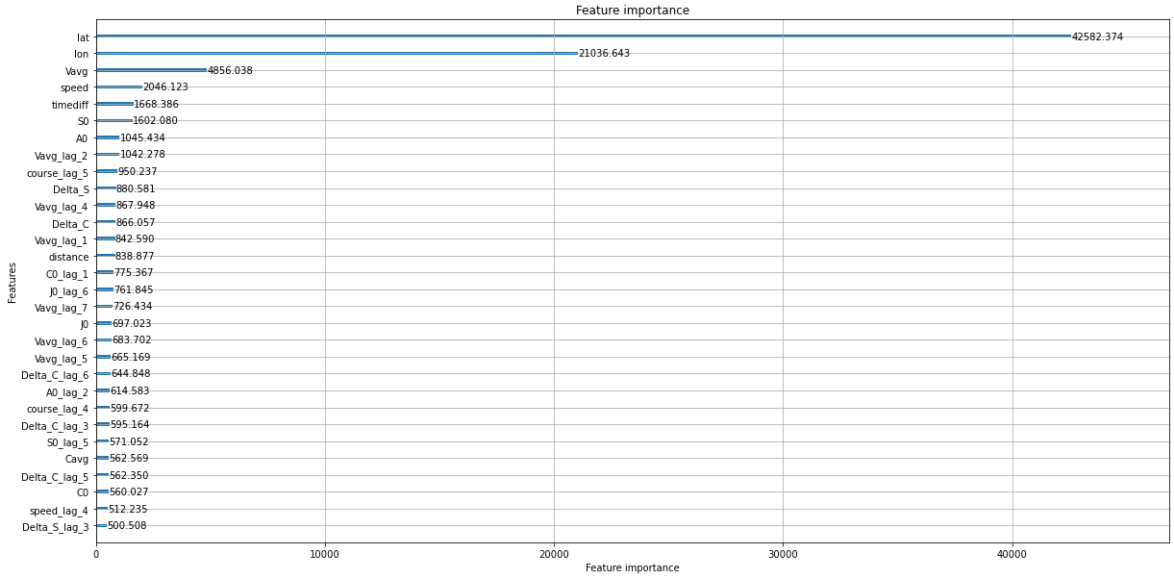

Figura 21. Importancia de las variables para Trawlers.

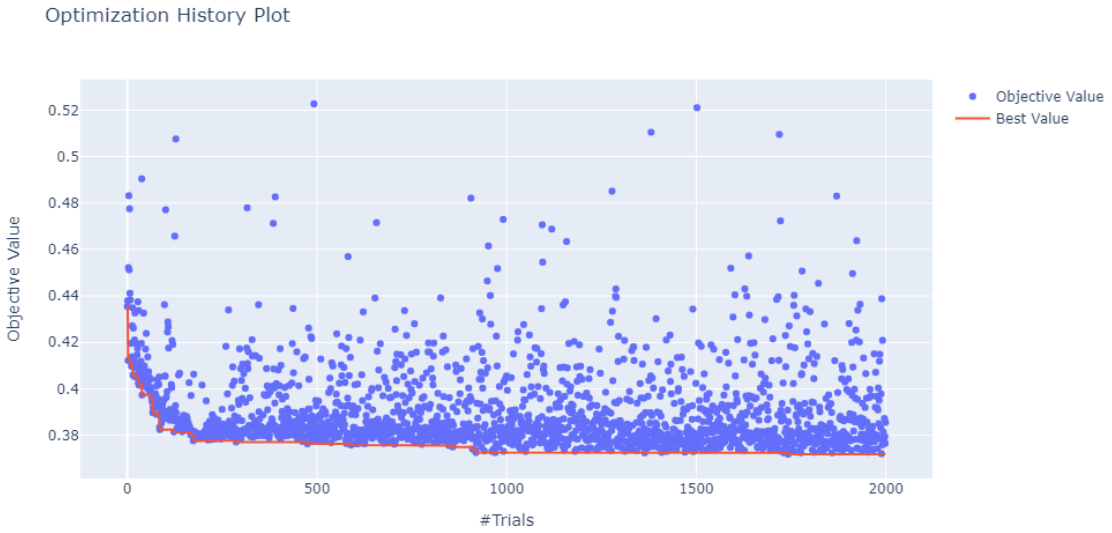

Figura 22. RMSE a lo largo del entrenamiento para Trawlers.

Trollers						
Modelo	Ajuste de Hiperparámetros	Hiperparámetros	Prec.	Rec.	Acc.	F1
Random Forest	GridSearch	• bootstrap: [True, False] • max_depth: [12,13,14,15,16,17,18,20,25] • max_features: ["auto", "sqrt", "log2"] • min_samples_leaf: [1,2,3,4,5,6,7,8] • min_samples_split: [2, 5 , 10, 20] • n_estimators: [100, 200,400,800,1000] • cv: 3	0.85	0.79	0.82	0.82
SVM sigmoid	GridSearch	• gamma: [1e-2,1e-3,1e-4,1e-5,1e-6] • C: [0.001,0.01,0.1,1,10,100] • coef0: [0.01,0.1,1,10]	0.68	0.66	0.67	0.67
SVM linear	GridSearch	• C: [0.0001,0.001,0.01,0.1,1,10,100,1000]	0.67	0.65	0.66	0.66
SVM rbf	GridSearch	• gamma:[1e-2,1e-3,1e-4,1e-5,1e-6] • C: [0.001,0.01,0.1,1,10,100]}	0.73	0.68	0.71	0.71
LGBM	Opt. Bayesiana	• Best value (rmse): 0.25440 • Best params: • n_estimators: 100 • learning_rate: 0.09906181756400692 • num_leaves: 118 • max_depth: 8 • min_data_in_leaf: 10 • lambda_l1: 0.00035699025524327244 • lambda_l2: 0.032656381277023125 • min_gain_to_split: 0.3045741702881734 • feature_fraction: 0.5	0.88	0.87	0.87	0.87

Tabla 10. Comparación de modelos para Trollers.

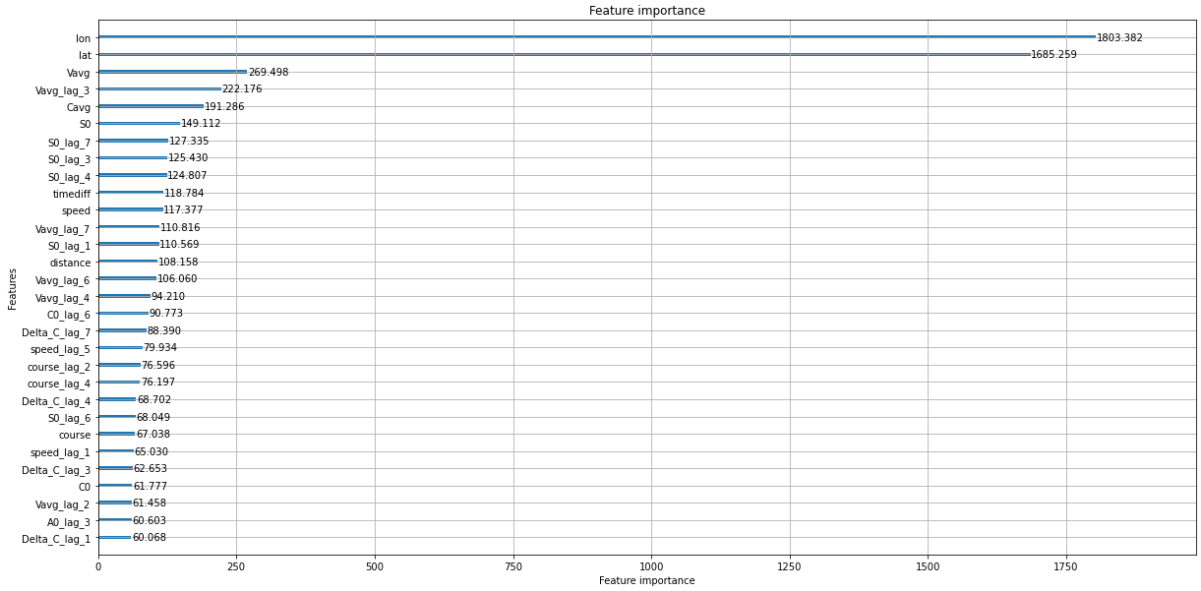

Figura 23. Importancia de las variables para Trollers.

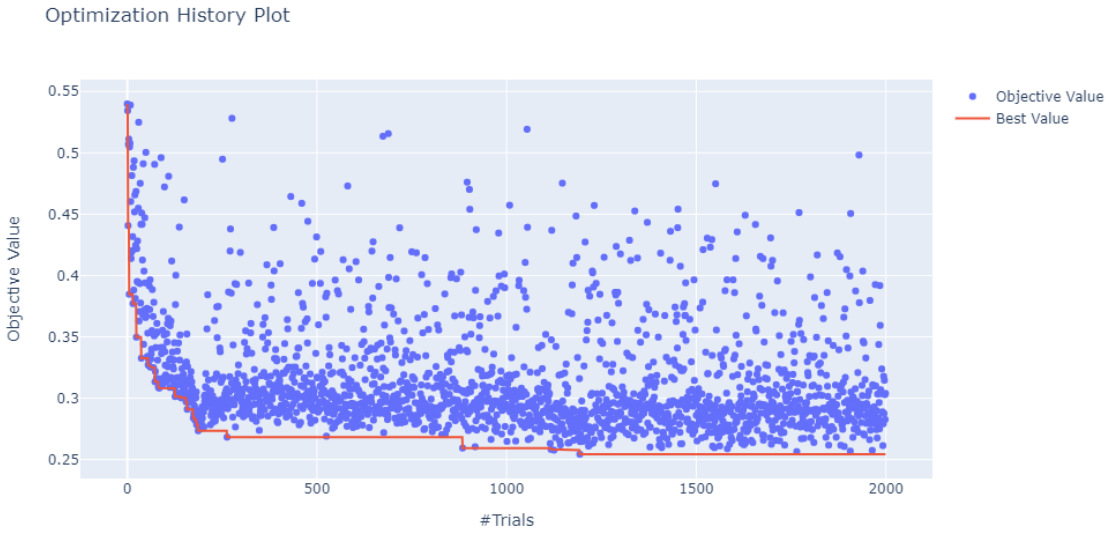

Figura 24. RMSE a lo largo del entrenamiento para Trollers.

8.6 Conclusiones

Clase de Buque	Precision	Recall	Accuracy	F1
Longlines	0.72	0.74	0.71	0.73
Purse Seines	0.77	0.67	0.75	0.72
Fixed Gear	0.736	0.729	0.726	0.732
Trawlers	0.82	0.82	0.82	0.82
Trollers	0.88	0.87	0.87	0.87

Tabla 11. Puntuación alcanzada con lightgbm.

Al criterio del autor han quedado pocos puntos para entrenar y estamos muy lejos de los F1 alcanzados por (Kroodsma et al., 2018). Hay que tener en cuenta que GFW solo comparte una parte del conjunto de datos que ellos utilizan para entrenar, y este es el único conjunto de datos disponible a nivel mundial etiquetado manualmente por expertos.
Los modelos utilizados con lightgbm alcanzaron buenos valores de RMSE, y mejores valores de F1 comparado con los métodos clásicos de minería de datos.
Como posible trabajo a futuro, al estar haciendo un submuestreo, podríamos utilizar más puntos de comparación brindando una salida de nuestro modelo donde sea un rango de tiempo para que detecte si el buque está pescando o no. Para ello nuestro conjunto de entrenamiento y testeo deberían de ser puntos consecutivos en el tiempo.
También podríamos entrenar el algoritmo de forma desbalanceada, ya que se trata de un problema desbalanceado, y configurar el parámetro “is_unbalance” de lightgbm para darle más peso a la clase minoritaria e intentar mejorar la performance de nuestro modelo.
Las dos variables más importantes han sido la latitud y longitud para nuestros modelos. Por lo tanto, si queremos saber si un buque se encuentra pescando en una zona la cual no corresponde al grupo de entrenamiento del modelo, la predicción no será tan buena. Podríamos probar de sacar estas variables, y que el modelo pondere más la importancia de las variables relativas al trayecto, y observar que sucede.

9. Bibliografía

Arasteh, S. T. (2020). Fishing Vessels Activity Detection from Longitudinal AIS Data. *GIS: Proceedings of the ACM International Symposium on Advances in Geographic Information Systems*, 347–356. doi:10.1145/3397536.3422267

De Souza, E. N. (2016). Improving fishing pattern detection from satellite AIS using data mining and machine learning. *PLoS ONE, 11*(7). doi:10.1371/JOURNAL.PONE.0158248

Fisheries and Oceans Canada. (12 de 9 de 2019). *Illegal, Unreported and Unregulated (IUU) Fishing*. Obtenido de https://www.dfo-mpo.gc.ca/international/isu-iuu-eng.htm

Global Fishing Watch. (25 de 7 de 2016). *Spoofing: One Identity Shared by Multiple Vessels*. Obtenido de https://globalfishingwatch.org/data/spoofing-one-identity-shared-by-multiple-vessels

Global Fishing Watch. (2020). *Anonymized AIS training data*. Obtenido de Datasets and Code: https://globalfishingwatch.org/data-download/datasets/public-training-data-v1

Global Fishing Watch. (2020). *What is AIS?* Obtenido de Frequently Asked Questions: https://globalfishingwatch.org/faqs/what-is-ais/

Hu, B. J. (2016). Identifying fishing activities from AIS data with Conditional Random Fields. *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems*, 47–52. doi:10.15439/2016F546

International Telecommunication Union. (2014). *Technical characteristics for an automatic identification system using time division multiple access in the VHF maritime mobile frequency band* (M Series ed.). Obtenido de https://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.1371-5-201402-I!!PDF-E.pdf

Jiang, X. S. (2016). Fishing activity detection from AIS data using autoencoders. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9673*, 33–39. doi:978-3-319-34111-8_4

Kroodsma, D. A. (s.f.). Tracking the global footprint of fisheries. *Science, 359*(6378), 904–908. doi:10.1126/science.aao5646

Microsoft Corporation. (2022). *Welcome to LightGBM's documentation*, dce7e58b. Obtenido de lightgbm: https://lightgbm.readthedocs.io/en/v3.3.2/

Shen, K. Y. (2020). A study of correlation between fishing activity and AIS data by deep learning. *TransNav, 14*(3), 527–531. doi:10.12716/1001.14.03.01